WE CLAIM:

Sub A41

1. A method for interfacing a first component and a second component, the first component being associated with a first electrical interface and communication type, and the second component being associated with a second electrical interface and communication type, comprising the following steps:

receiving from the first component a request for an access operation comprising one of at least a write operation and a read operation, and comprising a requested data address associated with the second component;

determining whether data corresponding to the requested data address is missing from memory, the memory being independent of the second component; and

when the data is missing from the memory, intercepting the request, loading the data from the second component to the memory, and performing the requested access operation, wherein modified data is converted from the first communication type to the second communication type and written to the memory and to the second component during a write operation, and wherein the data is converted from the second communication type to the first communication type and read by the first component during a read operation.

2. The method according to claim 1, wherein the step of determining whether data is missing from memory comprises determining whether the data is in an associated cache memory, and wherein the step of intercepting the request and loading the data comprises generating an interrupt request and writing the data to the associated cache memory, wherein the request from the first component is detained in response to the interrupt request while the data is being written to the associated cache memory.

3. The method according to claim 2, wherein the step of determining whether the data is stored in an associated cache memory comprises comparing the requested data address with addresses of stored data currently in the associated cache memory.

4. The method according to claim 2, wherein the step of writing the data to the associated cache memory comprises writing the data to a portion of the associated cache memory associated with stored data that is older than stored data associated with other portions of the associated cache memory.

5. The method according to claim 4, wherein the associated cache memory comprises cylinder random access memory and wherein the step of writing the data to a portion of the associated cache memory comprises writing the data to a page in the cylinder random access memory that is older than other pages in the cylinder random access memory.

6. The method according to claim 5, wherein the step of writing the data to a page comprises the steps of:

determining which of the pages in the cylinder random access memory is oldest;

determining which tracks of data within the oldest page have been modified;

writing the tracks of data which within the oldest page have been modified to the second component;

reading the data from the second component; and

writing the data to the oldest page.

7. The method according to claim 1, wherein the first communication type comprises a serial stream format and the second communication type comprises a parallel stream format.

8. The method according to claim 1, wherein during a write operation the modified data comprises a serial data stream, further comprising the steps of:

synchronizing a clock signal with the serial data stream;

shifting the synchronized serial data stream into a serial-to-parallel register; and

shifting a parallel data stream out of the register, wherein the serial-to-parallel register converts the serial data stream to the parallel data stream.

9. The method according to claim 1, wherein during a read operation the data comprises a parallel data stream,

shifting the parallel data stream into a parallel-to-serial register; and

shifting a serial data stream out of the register, wherein the parallel-to-serial register converts the parallel data stream to the serial data stream.

10. A method for writing a serial data stream to a component associated with parallel data streams, comprising the steps of:

receiving a serial data stream;

converting the serial data stream to a parallel data stream;

loading the parallel data stream in memory; and

writing the parallel data stream from the memory to a component.

11. The method according to claim 10, wherein the component comprises an integrated device electronics hard disk and wherein the step of receiving a serial data stream comprises receiving a serial data stream encoded according to the modified frequency modulation specification.

12. The method according to claim 10, wherein the step of converting the serial data stream to a parallel data stream comprises the steps of:

synchronizing a clock signal with the serial data stream;

shifting the synchronized data stream into a serial-to-parallel register; and

shifting data out of the register, wherein the data shifted out of the register comprises the parallel data stream.

13. The method according to claim 10, wherein the step of loading the parallel data stream in memory comprises loading the parallel data stream in cache memory.

14. The method according to claim 10, wherein the step of writing the parallel data stream to a component comprises the step of using a processor to write the parallel data

stream to the component, wherein the processor writes the parallel data stream to the component during a time frame that is convenient for the processor.

15. The method according to claim 10, further comprising the steps of:

identifying the parallel data written to the memory as being modified; and

writing the modified parallel data from the memory to the component when the component is idle.

16. A method for reading a serial data stream from a component associated with parallel data streams, comprising the steps of:

receiving a request for data associated with the component;

receiving a parallel data stream from memory in response to the request for the data;

converting the parallel data stream to a serial data stream; and

outputting the serial data stream.

17. The method according to claim 16, wherein the component comprises an integrated device electronics hard disk and wherein the step of converting the parallel data stream comprises converting the parallel data stream to a serial data stream encoded according to the modified frequency modulation specification.

18. The method according to claim 16, wherein the step of converting the parallel data stream to a serial data stream comprises the steps of:

shifting the parallel data stream into a parallel-to-serial register; and

shifting the serial data stream out of the register.

19. The method according to claim 16, wherein prior to the step of receiving a parallel data stream from memory, further comprising the steps of:

determining whether the data is missing from the memory, the memory being independent of the component; and

when the requested data is missing from the memory, interrupting the method, loading the data from the component to the memory, and resuming the method.

20. The method according to claim 19, wherein the step of determining whether the data is missing from the memory comprises determining whether the data is in cache memory, and wherein the steps of interrupting the method and loading the data from the component to the memory comprises generating an interrupt request and writing the data to the cache memory, wherein the request is detained in response to the interrupt request while the data is being written to the cache memory.

21. The method according to claim 10, wherein the step of loading the parallel data stream in memory comprises autonomously loading the parallel data stream in memory, wherein the parallel data stream is loaded without processor intervention.

22. The method according to claim 12, wherein the step of synchronizing the serial data stream comprises sampling the serial data stream at a rate at least twice that of a data rate of the serial data stream.

23. The method of claim 22, wherein the step of sampling the serial data stream comprises sampling the serial data stream according to a 10 MHZ clock.

24. The method of claim 15, wherein the serial data stream is associated with a host system, further comprising generating an interrupt request when the host system does not need to access the component, wherein the modified parallel data can then be written from the memory to the component.

25. The method of claim 20, wherein the step of generating an interrupt request further comprises the step of using a handshake signal line to detain the request in response to the interrupt request.

26. The method of claim 25, wherein the handshake signal line comprises a seek complete line.

27. The method of claim 19, wherein the step of loading the data comprises using a processor to read the data from the component and to write the data to the memory.

28. The method of claim 26, wherein the step of resuming the method comprises releasing the seek complete line, wherein the requested data is then autonomously read from the memory by the component.

29. The method of claim 16, wherein the step of outputting the serial data stream comprises repeatedly outputting the data, wherein an index pulse is outputted after each track of the data is outputted.

30. The method of claim 16, wherein the step of outputting the serial data comprises outputting the serial data stream without processor intervention.

31. The method of claim 19, wherein the steps of converting the parallel data stream and outputting the serial data stream are performed substantially immediately after a determination that the data is in the memory.

32. The method of claim 19, wherein the step of loading the data comprises writing the data to a portion of the memory that is older than other portions of the memory.

33. The method of claim 32, further comprising the step of writing tracks of data within the oldest portion of the memory to the component prior to writing the data to the oldest portion.

34. The method of claim 1, wherein the step of performing the requested access operation comprises performing the access operation autonomously, whereby the access operation is performed without processor intervention.

35. A computer readable medium, comprising instructions capable of performing the method of claim 1.

36. A computer readable medium, comprising instructions capable of performing the method of claim 10.

37. A computer readable medium, comprising instructions capable of performing the method of claim 16.